# Hagelslag: Scalable Object-Based Severe Weather Analysis and Forecasting

David John Gagne II[1,3,4], Amy McGovern[2], Nate Snook[1], Ryan Sobash[3], Jon Labriola[1,4], John K. Williams[5], Sue E. Haupt[3], Ming Xue[1,4]

1. Center for Analysis and Prediction of Storms, University of Oklahoma (OU), 2. School of Computer Science, OU,

3. National Center for Atmospheric Research, 4. School of Meteorology, OU 5. Weather Company/WSI

## Description

*Ha•gel•slag (n.)*

1. Dutch word for hailstorm
2. A Python package for storm-based analysis, forecasting, and evaluation
3. Dutch breakfast item consisting of sprinkles and butter on white bread

Seriouseats.com

### Features

- Modular: self-contained modules
- Scalable: parallel processing of model runs
- Fast: smart algorithm design + SciPy stack
- Open-Source: available through Github and PyPI

## Package Information

### Dependencies

- Numpy, Scipy, Matplotlib, Pandas
- Scikit-learn, Scikit-image
- netCDF4-python, Basemap

### Data Management

Hagelslag supports reading gridded model and observations in netCDF format. A generic ModelGrid class can be extended to handle other model data structures. Currently the CAPS and NCAR Ensembles are directly supported. Storm track information is output in geoJSON and csv formats, and evaluation data is stored in csv format.

### Applications

*hsdata:* Loads forecast and observations, finds storm tracks, matches tracks, extracts storm attributes.
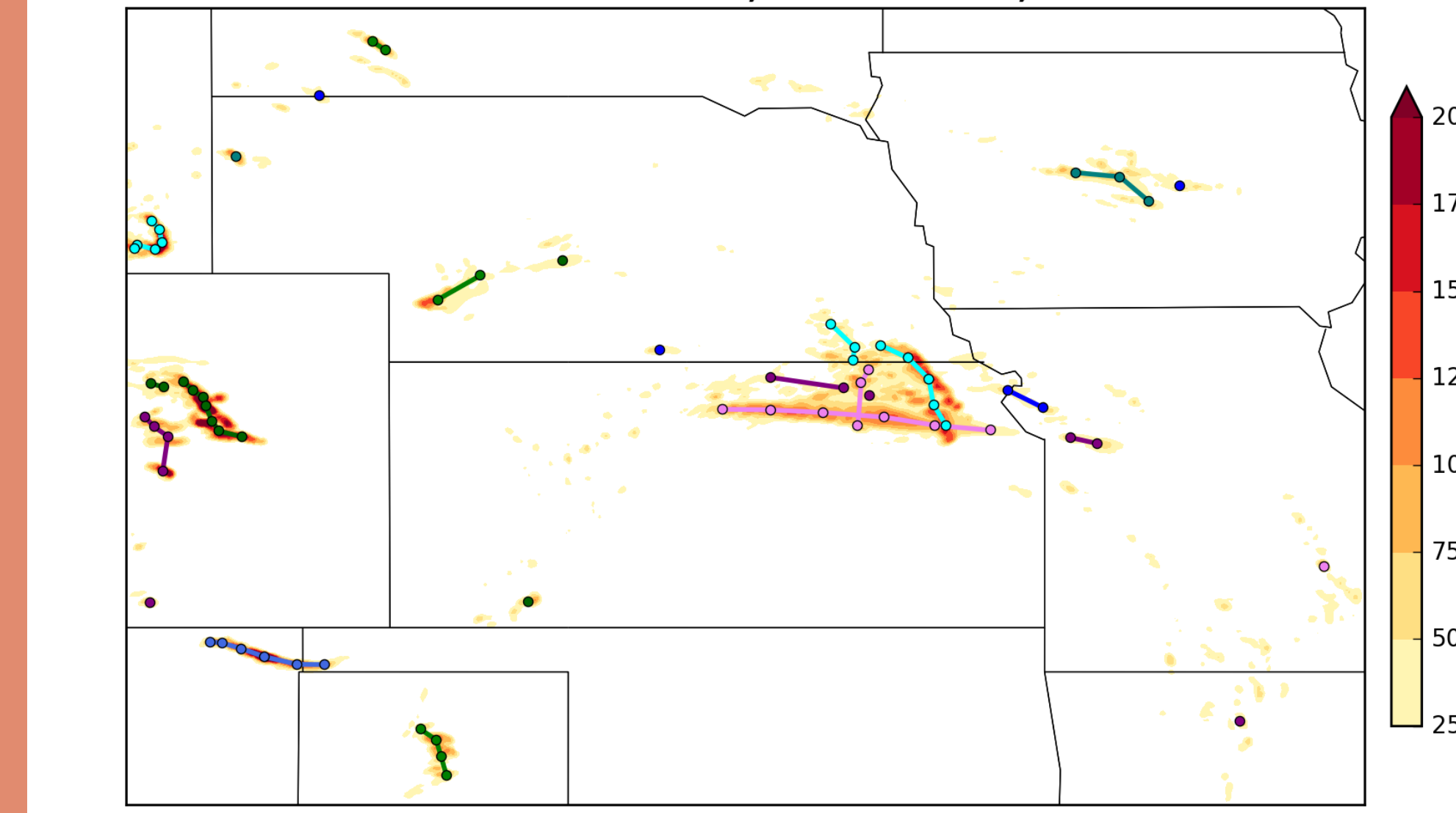
*hsforecast:* Trains and runs machine learning models, generates ensemble forecast products, including neighborhood probabilities.

*hseval:* Performs evaluation of gridded and object-based forecasts.

*hsviewer:* Interactive storm track viewer.

## Storm Tracking and Matching



Forecast Tracks Shifted Centroid: 1.0, Centroid: 0.0, Max Distance: 50 km

Hagelslag storm tracks overlaid on maximum updraft helicity over the full day.

Identify objects from grids with Enhanced Watershed at each time step.

Link objects into tracks using a custom distance function and optimal matching.

Match forecast and observed tracks using a custom distance function.
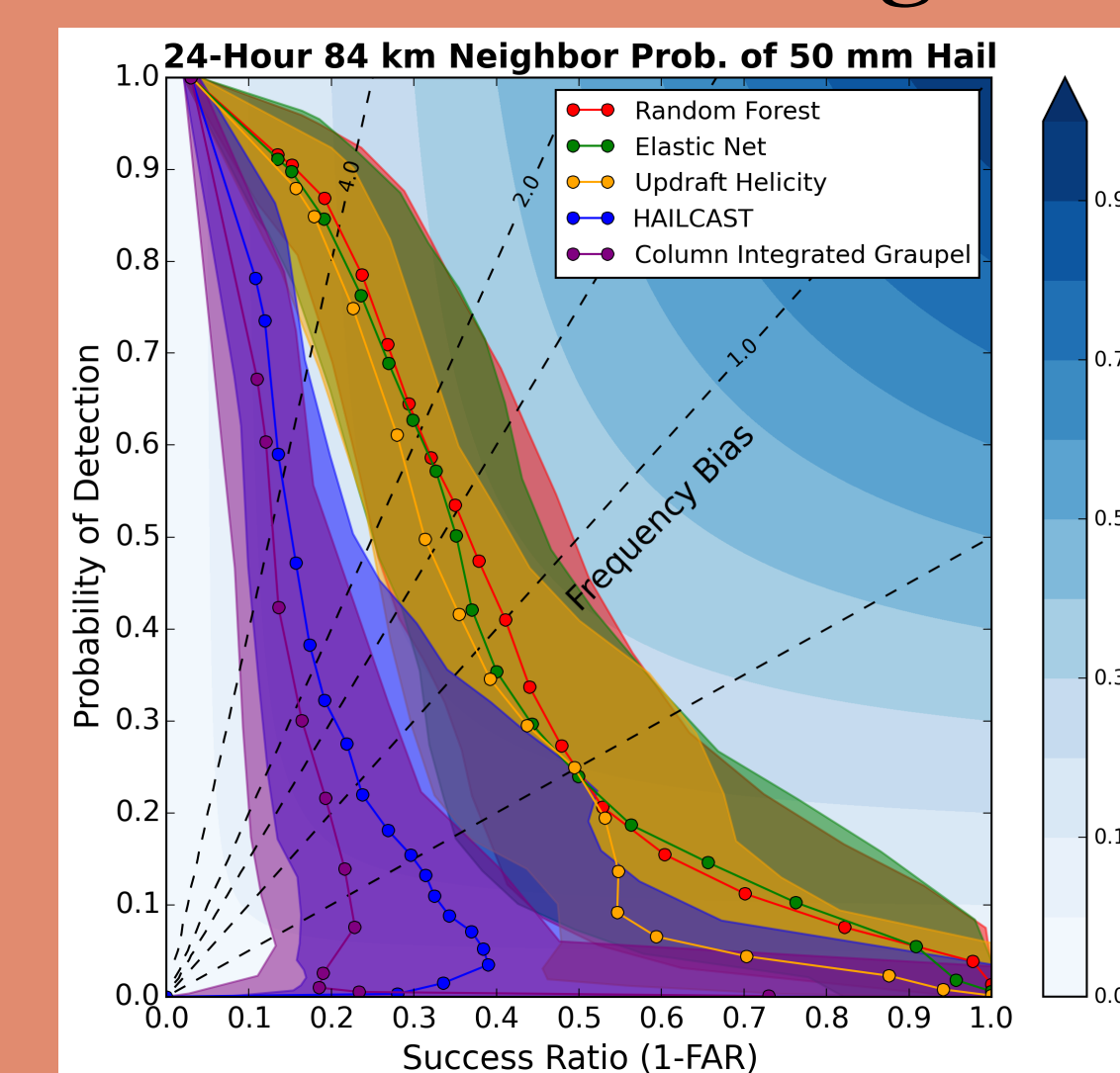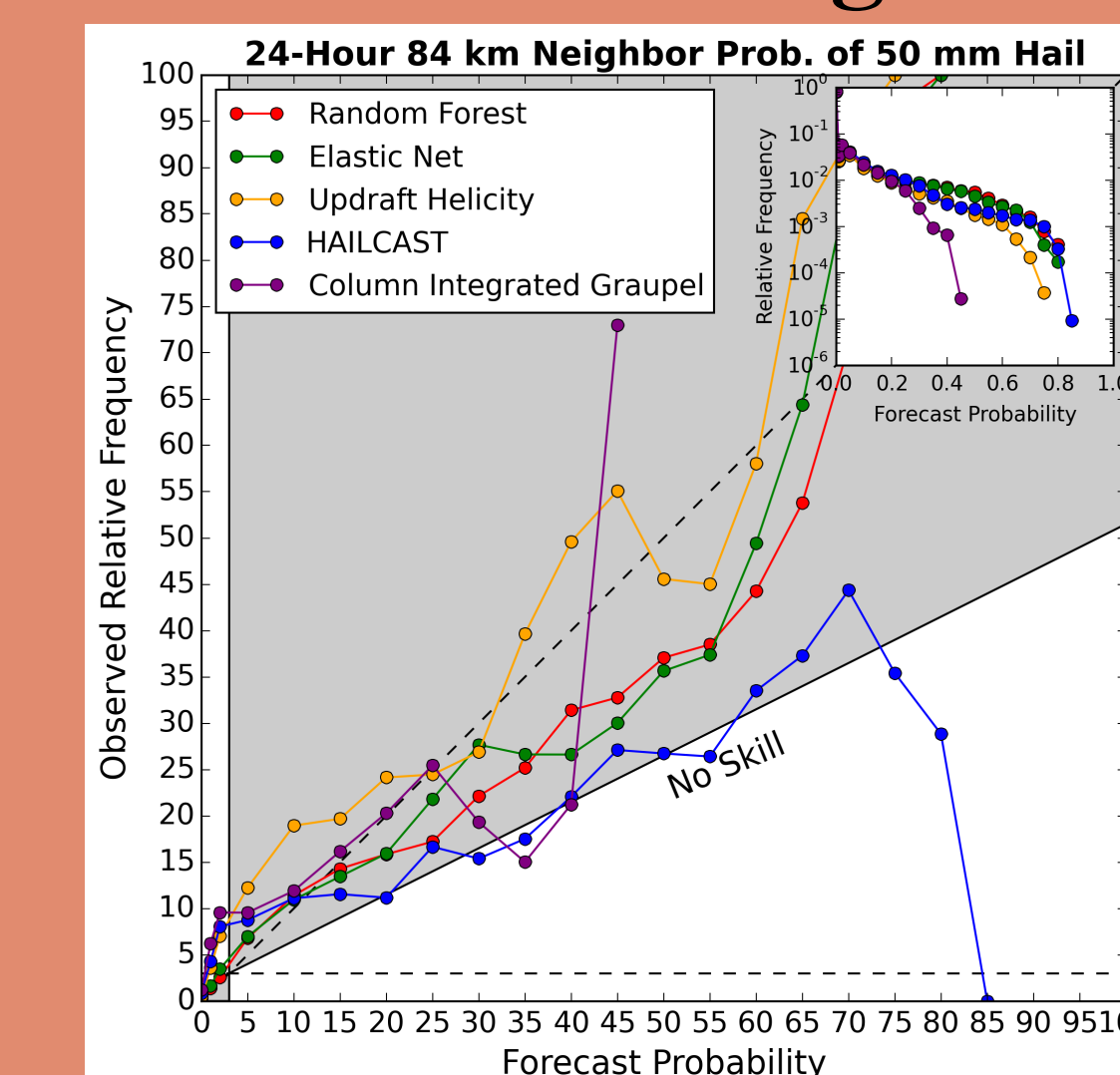
**Example applications**
Better storm motion estimates
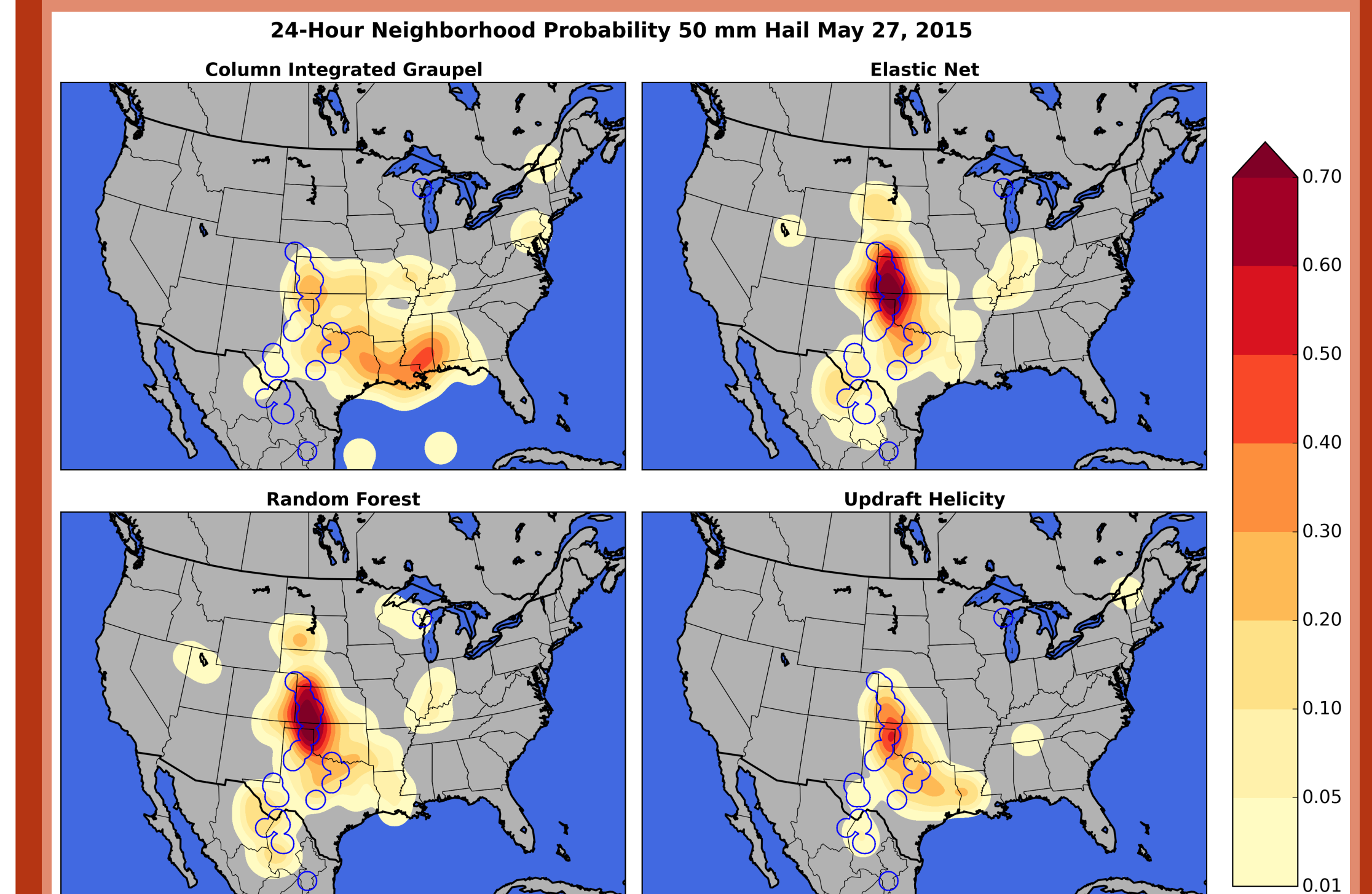Storm proxy climatologies

```
match= 0.5 * start_centroid_distance +
       0.3 * start_time_distance +
       0.2 * mean_area_distance
```

## Distributed Probability Forecast Evaluation

Split forecasts into independent components

Evaluate Component 1

Evaluate Component 2

...

Evaluate Component N

Discretize forecasts into probability bins and save relevant bin counts and total number of forecasts as DistributedROC and DistributedReliability objects

Sum relevant forecast counts together

Bootstrap resample object sets and sum

Calculate summary statistics, diagrams

Calculate confidence intervals

### Performance Diagram



### Attributes Diagram



## Post-Processing



24-Hour Neighborhood Probability 50 mm Hail May 27, 2015

Hagelslag trains and runs machine learning models from scikit-learn. Custom configurations can be specified in the hsforecast config file.

Hagelslag can generate neighborhood ensemble probabilities for both raw model variables and machine learning model predictions.

## Distribution

**Github**
https://github.com/djgagne/hagelslag

**Python Package Index**
https://pypi.python.org/pypi/hagelslag

**Contact Information**
Email: djgagne@ou.edu
Twitter: @DJGagneDos

## Acknowledgements